IMPROVED TECHNIQUES FOR TRAINING SMALLER AND FASTER STABLE DIFFUSION

Hesong Wang, Huan Wang
Westlake University
{wanghesong, wanghuan}@westlake.edu.cn

ABSTRACT

Recent SoTA text-to-image diffusion models achieve impressive generation quality but their computational cost has been prohibitively large. Network pruning and step distillation are two widely-used compression techniques to reduce the model size and inference steps. This work presents a few improved techniques in these aspects to train smaller and faster diffusion models with a cheap training cost. Specifically, compared to the prior SoTA counterparts, we introduce a structured pruning method to remove insignificant weight blocks based on an improved performance sensitivity. To regain performance after pruning, a CFG-aware retraining loss is proposed, which is shown critical to performance. Finally, a modified CFG-aware step distillation is used to reduce the steps. Empirically, our method manages to prune the U-Net parameters of SD v2.1 base by 46%, inference steps reduced from 25 to 8, achieving an overall $3.0\times$ wall-clock inference speedup in 512×512 image generation. Our 8-step model is significantly better than 25-step BK-SDM, the prior SoTA for cheap Stable Diffusion, while being even smaller.

1 INTRODUCTION

Generative models have seen tremendous growth in recent years, with diffusion models emerging as one of the most successful techniques, particularly in text-to-image generation. These models, such as DDPM (Ho et al., 2020), DDIM (Song et al., 2020), and LDM (Rombach et al., 2022), have shown impressive performance in generating high-quality images from text descriptions. However, diffusion models, especially those based on the U-Net architecture, come with a significant computational burden. The multi-step denoising process required for high-quality generation makes these models highly resource-intensive, thus limiting their application in resource-constrained environments.

Therefore, many works seek model compression techniques to make diffusion models more efficient. One primary group of methods is to use network pruning and distillation, such as knowledge distillation (Hinton et al., 2014) for transferring model capacity to smaller networks, which is represented by SnapFusion, MobileDiffusion, and BK-SDM. Among them, BK-SDM is featured by their *cheap* compression cost (4-13 A100 days) while the other two spend hundreds A100 days by estimation. In this work, we advance this technical path by presenting improved techniques of pruning and distillation to train efficient stable diffusion models, within a low training budget.

Specifically, we introduce a structured pruning method based on performance sensitivity analysis, which is shown to prune more accurately than BK-SDM. Besides, for the retraining part, we tap into the idea of CFG-aware step distillation (Li et al., 2024) to introduce a CFG-aware training loss function, which is shown to be critical to fast convergence during retraining. In addition, we also introduce a step reduction training technique that minimizes the number of denoising steps during inference, which can reduce the inference steps to 8 steps with marginal quality degradation.

Empirically, with the improved pruning and distillation techniques, our method manages to prune the U-Net parameters of SD v2.1 base from 866M to 466M (46% reduction), inference steps reduced from 25 to 8, achieving an overall $3.0\times$ wall-clock inference speedup in 512×512 text-to-image generation on an A6000 GPU. Compared to the SoTA counterpart BK-SDM, our 8-step model is significantly better than 25-step BK-SDM in the generation quality, while being even smaller.

The contributions of this paper are threefold:

- We propose a performance-sensitivity pruning strategy that structurally prunes the U-Net in SD, which is more accurate than that proposed in the prior SoTA of this line (BK-SDM).
- We introduce a CFG-aware training loss that effectively regains the model performance after pruning by utilizing classifier-free guidance to improve the quality of features during training.
- We show the proposed techniques are very easy to be integrated into the existing SD training pipeline. Empirically, the overall scheme prunes the U-Net parameters of SD v2.1 base from 866M to 466M (46% reduction), inference steps reduced from 25 to 8, reporting $3\times$ wall-clock speedup, and significantly surpassing the prior SoTA counterpart BK-SDM.

2 Related Work

Since this work aims for efficient diffusion models by using network pruning and step distillation. Here we include the most relevant papers in these two axes.

Network pruning of diffusion models. The computational burden of diffusion models, particularly U-Net, has led to efforts in model compression and pruning (Han et al., 2016; Wang et al., 2021). SnapFusion (Li et al., 2024), BK-SDM (Kim et al., 2024), and MobileDiffusion (Zhao et al., 2023) focus on pruning less essential components of the U-Net, following classical deep compression and structured pruning frameworks that preserve important weight groups. These methods improve efficiency by removing redundant parameters, reducing both memory footprint and inference time.

Step reduction of diffusion models. A major challenge in diffusion models is the number of denoising steps required for high-quality generation. Progressive Distillation (Salimans & Ho, 2022), SnapFusion (Li et al., 2024), Consistency Models (Song et al., 2023), LCM (Luo et al., 2023), and InstaFlow (Liu et al., 2023) reduce denoising steps while maintaining generative quality. More aggressive methods like DMD (Yin et al., 2024b), DMD2 (Yin et al., 2024a), UFOGen (Xu et al., 2024), ADD (Sauer et al., 2024b), and LADD (Sauer et al., 2024a) aim for single-step image generation, significantly improving efficiency and enabling real-time generation with good quality.

3 Proposed Method

Our method primarily consists of three components:

- **Performance-sensitivity pruning** reduces the U-Net model size by selectively pruning less critical blocks based on a more accurate sensitivity analysis.
- **CFG-aware retraining** improves the training quality by enhancing the alignment between the pruned and original models, helping recover from pruning.
- **Step distillation** reduces the number of inference steps by building upon the CFG-aware step distillation proposed in SnapFusion (Li et al., 2024), with a few modifications for improvement.

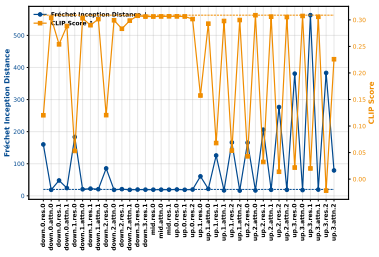

Figure 1: Pruning sensitivity analysis of SD v2.1 base at 25 steps. The x-axis represents the pruned blocks in the U-Net. The orange and blue dashed lines represent the CLIP score and FID of unpruned SD v2.1 base, respectively.

3.1 Performance-sensitivity pruning

Prior pruning methods struggle to balance simplicity and efficiency. They typically require learning the pruned structure in a training process (Wang et al., 2021), or demand extensive retraining to restore model performance. To overcome these limitations, we propose a block-wise pruning approach that directly evaluates each block's sensitivity without additional training overhead. By identifying and removing the least critical blocks, our method significantly reduces model complexity while preserving generative quality.

U-Net consists of multiple ResNet and attention blocks. We conduct block-wise sensitivity analysis to determine the least critical components. Specifically, we replace individual blocks with identity mappings and evaluate the model's performance each time. Blocks with minimal contribution to overall performance are pruned to reduce model size and computational cost.

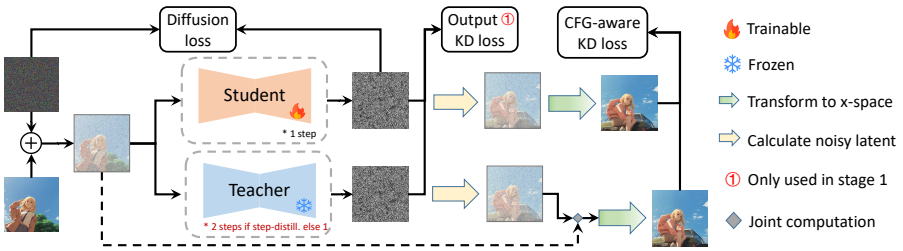

Figure 2: Overview of the two-stage training in our method. The first-stage training is for regaining performance after pruning; the second-stage training is to reduce inference steps. Three losses are involved: diffusion loss, output KD loss (only used during the first stage), and CFG-aware KD loss.

Fig. 1 presents the performance evaluation of the pruned U-Net. This analysis ensures that our pruning strategy eliminates non-essential blocks while maintaining high generation quality.

A key advantage of our approach is its simplicity and intuitiveness. Unlike methods that require complex training-based pruning mechanisms, our strategy directly assesses the importance of each block without requiring the training process. This allows for an efficient pruning procedure that avoids extensive retraining, making it highly practical for real-world applications. Furthermore, the straightforward nature of block-wise sensitivity analysis ensures easy integration into existing diffusion model pipelines with minimal overhead.

3.2 Stage 1: CFG-aware Retraining

Inspired by SnapFusion (Li et al., 2024) and classic (feature) distillation works (Hinton et al., 2014; Romero et al., 2015; Wang et al., 2020), we introduce a CFG-aware training loss designed to enhance the training feature quality, which in turn improves the final image generation quality. Our motivation stems from the fact that CFG has been shown to significantly improve the quality of generated images by guiding the diffusion process in a more controlled manner. Note, SnapFusion introduced the CFG-aware training for reducing inference steps, while here we show it can also be used for improving the retraining stage of network pruning.

Specifically, the CFG-aware training loss operates by first applying CFG to the outputs of both the teacher and student models. Then, the outputs are converted into clean latents in the x-space. Finally, we compute the MSE loss between the clean latents of the teacher and student models:

$$\mathbf{x}(\theta, \mathbf{z}_t, t, \mathbf{c}) = \frac{\mathbf{z}_t - \sqrt{1-\bar{\alpha}_t}\,\mathrm{CFG}(\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}))}{\sqrt{\bar{\alpha}_t}}, \quad \mathcal{L}_{\mathrm{CfgKD}} = \mathbb{E}\left[\|\mathbf{x}(T, \mathbf{z}_t, t, \mathbf{c}) - \mathbf{x}(S, \mathbf{z}_t, t, \mathbf{c})\|_2^2\right], \tag{1}$$

where $\mathbf{x}$ represents the predictor to obtain the clean latents, and ϵ means the noise predictor.

In addition to this retraining loss, we also employ the output alignment loss $\mathcal{L}_{\mathrm{OutKD}}$ and the original diffusion denoising loss $\mathcal{L}_{\mathrm{Diff}}$, the overall training objective in this phase is:

$$\mathcal{L}_{\mathrm{Stage1}} = \lambda_{\mathrm{Diff}}\mathcal{L}_{\mathrm{Diff}} + \lambda_{\mathrm{OutKD}}\mathcal{L}_{\mathrm{OutKD}} + \lambda_{\mathrm{CfgKD}}\mathcal{L}_{\mathrm{CfgKD}}, \mathcal{L}_{\mathrm{OutKD}} = \mathbb{E}\left[\|\epsilon_T(\mathbf{z}_t, t, \mathbf{c}) - \epsilon_S(\mathbf{z}_t, t, \mathbf{c})\|_2^2\right]. \tag{2}$$

This loss aims to align the output distributions of the student and teacher models, ensuring that the pruned model retains the generalization capability of the full model.

3.3 Stage 2: Step Distillation

Here we further reduce the number of inference steps based on CFG-aware distillation method proposed in SnapFusion (Li et al., 2024).

In SnapFusion, the coefficient before the CFG-ware distillation loss term is set to 0.2, while in our case, we empirically find using a larger weight for this term is more beneficial to performance, so we set the coefficient to 1. We also save the truncated SNR reweighting in SnapFusion.

The overall loss function in this stage is

$$\mathcal{L}_{\mathrm{Stage2}} = \lambda_{\mathrm{Diff}}\mathcal{L}_{\mathrm{Diff}} + \lambda_{\mathrm{StepReduc}}\mathcal{L}_{\mathrm{StepReduc}}, \quad \mathcal{L}_{\mathrm{StepReduc}} = \mathbb{E}\left[\left\|\mathbf{x}(S, \mathbf{z}_t, t, \mathbf{c}) - \frac{\mathbf{z}_{t''} - \frac{\sigma_{t''}}{\sigma_t}\mathbf{z}_t}{\alpha_{t''} - \frac{\sigma_{t''}}{\sigma_t}\alpha_t}\right\|_2^2\right], \tag{3}$$

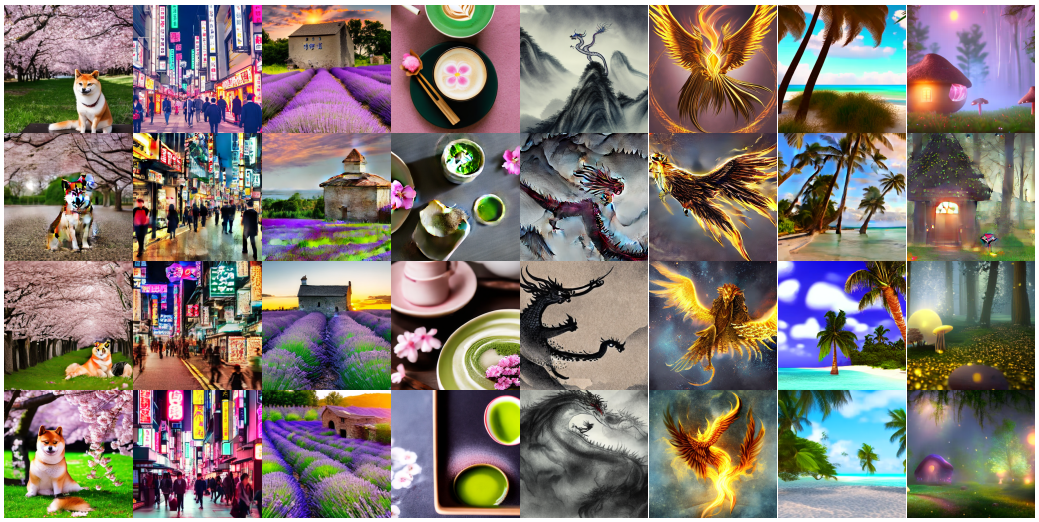

Figure 3: Visual results comparison of different models: *Row 1:* SD v2.1 base (25 steps). *Row 2:* SD v2.1 base (8 steps). *Row 3:* BK-SDM v2 base (25 steps). *Row 4:* Our pruned mode (8 steps).

Table 1: Comparison of various models on generation quality with their number of parameters. Quality scores are evaluated on 5K samples of MS COCO 2014.

Model		**# Param (M)**		**Quality Score**		
Name	# Steps	Total	U-Net	FID↓	CLIP score↑	IS↑
SD v2.1 base	25	1256	866	19.31	0.3064	32.06
SD v2.1 base	8	1256	866	29.39	0.2721	22.14
BK-SDM v2 base	25	973	583	21.32	0.2878	28.65
BK-SDM v2 base	8	973	583	38.46	0.2415	17.95
Ours (pruned w. fine-tuning)	25	856	466	19.10	**0.3017**	**29.11**
Ours (pruned w. fine-tuning + step distill.)	**8**	**856**	**466**	**18.07**	0.2960	28.02

where $\mathbf{z}_{t''}$ is the teacher's noisy latent with two denoising steps.

The overall training loss combinations are depicted in Fig. 2.

4 Experimental Results

Experiment setups. We conducted our experiments using 8 NVIDIA A100 GPUs (total batch size 512). The dataset used for training was a subset of the LAION dataset, containing approximately 2 million text-image pairs. We employed a learning rate of 5.0×10^{-5} with AdamW optimizer.

Performance comparison. Tab. 1 shows the results of our model compared with others. (1) Our proposed model reduces the number of parameters in the U-Net from 866M to 466M parameters (46% reduction), achieving comparable scores while using only 8 steps. (2) Compared to BK-SDM, the prior SoTA in this line for training cheap SD, our 8-step pruned model performs significantly better than the 25-step BK-SDM model while being even smaller in model size.

Training and inference speedup analysis. (1) By our record, early in the training process, our model *outperforms* the BK-SDM v2 base after only 2K iterations, equivalent to approximately **one A100 day**, while BK-SDM v2 base takes **4 A100 days** for training, *i.e.*, our method achieves $4\times$ training speedup. (2) Our pruned model achieves 512×512 text-to-image generation in 0.31s on an A6000 GPU, while the original SD v2.1 base takes 0.92s, *i.e.*, $3\times$ inference speedup.

5 Conclusion

This work proposes improved network pruning and step distillation techniques for training smaller and faster Stable Diffusion models at low cost. We introduce a performance-sensitivity-based block pruning method, a CFG-aware retraining loss to recover performance, and a modified step distillation approach to reduce inference steps. Overall, our method prunes 46% of U-Net parameters in SD v2.1 base, achieves a $3\times$ wall-clock speedup in 512×512 text-to-image generation, and significantly outperforms the prior SoTA BK-SDM despite being smaller.

REFERENCES

Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *4th International Conference on Learning Representations (ICLR)*, San Juan, Puerto Rico, May 2016. URL https://arxiv.org/abs/1510.00149.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *Neural Information Processing Systems (NIPS) Deep Learning Workshop*, Montréal, Canada, December 2014. URL https://arxiv.org/abs/1503.02531.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Bo-Kyeong Kim, Hyoung-Kyu Song, Thibault Castells, and Shinkook Choi. Bk-sdm: A lightweight, fast, and cheap version of stable diffusion. In *European Conference on Computer Vision*, pp. 381–399. Springer, 2024.

Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Snapfusion: Text-to-image diffusion model on mobile devices within two seconds. *Advances in Neural Information Processing Systems*, 36, 2024.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*, 2023.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024a.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024b.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Huan Wang, Yijun Li, Yuehai Wang, Haoji Hu, and Ming-Hsuan Yang. Collaborative distillation for ultra-resolution universal style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Huan Wang, Can Qin, Yulun Zhang, and Yun Fu. Neural pruning via growing regularization. In *International Conference on Learning Representations (ICLR)*, 2021.

Yanwu Xu, Yang Zhao, Zhisheng Xiao, and Tingbo Hou. Ufogen: You forward once large scale text-to-image generation via diffusion gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8196–8206, 2024.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *arXiv preprint arXiv:2405.14867*, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6613–6623, 2024b.

Yang Zhao, Yanwu Xu, Zhisheng Xiao, and Tingbo Hou. Mobilediffusion: Subsecond text-to-image generation on mobile devices. *arXiv preprint arXiv:2311.16567*, 2023.